

Docket No.: 57454-206

PATENT

#3

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of

Yasuhiro NUNOMURA

Serial No.: Group Art Unit:

Filed: August 14, 2001 Examiner:

For: MULTIPROCESSOR SYSTEM CONTROLLING FREQUENCY OF CLOCK INPUT
TO PROCESSOR ACCORDING TO RATIO OF PROCESSING TIMES OF
PROCESSORS, AND METHOD THEREOF

CLAIM OF PRIORITY AND
TRANSMITTAL OF CERTIFIED PRIORITY DOCUMENT

Commissioner for Patents
Washington, DC 20231

Sir:

In accordance with the provisions of 35 U.S.C. 119, Applicant hereby claims the priority of:

Japanese Patent Application No. 2001-009802, filed January 18, 2001

cited in the Declaration of the present application. A Certified copy is submitted herewith.

Respectfully submitted,

MCDERMOTT, WILL & EMERY


Stephen A. Becker
Registration No. 26,527

600 13th Street, N.W.
Washington, DC 20005-3096
(202) 756-8000 SAB:prp
Date: August 14, 2001
Facsimile: (202) 756-8087

日本国特許庁

PATENT OFFICE
JAPANESE GOVERNMENT

57454-206
NONOMURA
August 14, 2001

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日

Date of Application:

2001年 1月18日

出願番号

Application Number:

特願 2001-009802

出願人

Applicant (s):

三菱電機株式会社

A standard linear barcode is positioned vertically on the left side of the page. It is oriented such that the vertical axis of the barcode is parallel to the page's edge.

2001年 2月 9日

特許庁長官
Commissioner,
Patent Office

及川耕造

出証番号 出証特2001-3006741

【書類名】 特許願
【整理番号】 528095JP01
【提出日】 平成13年 1月18日
【あて先】 特許庁長官殿
【国際特許分類】 G06F 15/16
G06F 1/32
G06F 1/04

【発明者】

【住所又は居所】 東京都千代田区丸の内二丁目2番3号 三菱電機株式会社内

【氏名】 布村 泰浩

【特許出願人】

【識別番号】 000006013

【氏名又は名称】 三菱電機株式会社

【代理人】

【識別番号】 100064746

【弁理士】

【氏名又は名称】 深見 久郎

【選任した代理人】

【識別番号】 100085132

【弁理士】

【氏名又は名称】 森田 俊雄

【選任した代理人】

【識別番号】 100091409

【弁理士】

【氏名又は名称】 伊藤 英彦

【選任した代理人】

【識別番号】 100096781

【弁理士】

【氏名又は名称】 堀井 豊

【選任した代理人】

【識別番号】 100096792

【弁理士】

【氏名又は名称】 森下 八郎

【手数料の表示】

【予納台帳番号】 008693

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 マルチプロセッサシステムおよびその制御方法

【特許請求の範囲】

【請求項1】 複数のプロセッサが並列に処理を行なった後に、同期を取るマルチプロセッサシステムであって、

第1のプロセッサと、

前記第1のプロセッサと異なる第2のプロセッサと、

前記第1のプロセッサの処理時間と前記第2のプロセッサの処理時間との比率に応じて、前記第1のプロセッサに入力されるクロックの周波数を制御するためのクロック周波数制御手段とを含む、マルチプロセッサシステム。

【請求項2】 前記マルチプロセッサシステムはさらに、前記第1のプロセッサに入力されるクロックの周波数に応じて、前記第1のプロセッサに供給される電源電圧を制御するための電圧制御手段を含む、請求項1記載のマルチプロセッサシステム。

【請求項3】 前記マルチプロセッサシステムはさらに、前記第1のプロセッサの処理時間見積値と前記第2のプロセッサの処理時間見積値とが予め格納される格納手段と、

前記格納手段に格納された第1のプロセッサの処理時間見積値と第2のプロセッサの処理時間見積値との比率に応じて前記クロック周波数制御手段を制御して、前記第1のプロセッサに入力されるクロックの周波数を変更するための周波数決定手段とを含む、請求項1または2記載のマルチプロセッサシステム。

【請求項4】 前記マルチプロセッサシステムはさらに、前記第1のプロセッサの処理時間見積値と前記第2のプロセッサの処理時間見積値との比率が予め格納される格納手段を含み、

前記格納手段に格納された比率に応じて前記クロック周波数制御手段を制御して、前記第1のプロセッサに入力されるクロックの周波数を変更するための周波数決定手段とを含む、請求項1または2記載のマルチプロセッサシステム。

【請求項5】 前記マルチプロセッサシステムはさらに、前回の処理において、前記第1のプロセッサの処理が前記第2のプロセッサの処理よりも先に完了

した場合には、前記クロック周波数制御手段に第1のプロセッサに入力されるクロックの周波数を低くさせ、

前回の処理において、前記第2のプロセッサの処理が前記第1のプロセッサの処理よりも先に完了した場合には、前記第1のプロセッサに入力されるクロックの周波数を高くさせるための周波数決定手段を含む、請求項1または2記載のマルチプロセッサシステム。

【請求項6】 前記マルチプロセッサシステムはさらに、前記第1のプロセッサの処理と前記第2のプロセッサの処理とのいずれが先に完了するかを予測するための予測手段と、

前記予測手段による予測結果に応じて前記クロック周波数制御手段を制御して、前記第1のプロセッサに入力されるクロックの周波数を変更するための周波数決定手段とを含む、請求項1または2記載のマルチプロセッサシステム。

【請求項7】 前記予測手段は、前回の処理において、前記第1のプロセッサの処理が前記第2のプロセッサの処理よりも先に完了した場合には、前記第1のプロセッサの方が先に処理を完了すると予測し、

前回の処理において、前記第2のプロセッサの処理が前記第1のプロセッサの処理よりも先に完了した場合には、前記第2のプロセッサの方が先に処理を完了すると予測する、請求項6記載のマルチプロセッサシステム。

【請求項8】 前記予測手段は、前回の処理において、前記第1のプロセッサの処理が前記第2のプロセッサの処理よりも先に完了した場合には、前記第2のプロセッサの方が先に処理を完了すると予測し、

前回の処理において、前記第2のプロセッサの処理が前記第1のプロセッサの処理よりも先に完了した場合には、前記第1のプロセッサの方が先に処理を完了すると予測する、請求項6記載のマルチプロセッサシステム。

【請求項9】 前記マルチプロセッサシステムはさらに、前記第1のプロセッサの処理時間を計測するための第1の計測手段と、

前記第2のプロセッサの処理時間を計測するための第2の計測手段と、

前記第1の計測手段による計測結果および前記第2の計測手段による計測結果に応じて前記クロック周波数制御手段を制御して、前記第1のプロセッサに入力

されるクロックの周波数を変更するための周波数決定手段とを含む、請求項1または2記載のマルチプロセッサシステム。

【請求項10】 複数のプロセッサが並列に処理を行なった後に、同期を取るマルチプロセッサシステムの制御方法であって、

第1のプロセッサの処理時間と第2のプロセッサの処理時間との比率を算出するステップと、

前記算出された比率に応じて、前記第1のプロセッサに入力されるクロックの周波数を制御するステップとを含む、マルチプロセッサシステムの制御方法。

【請求項11】 前記制御方法はさらに、前記第1のプロセッサに入力されるクロックの周波数に応じて、前記第1のプロセッサに供給される電源電圧を制御するステップを含む、請求項10記載のマルチプロセッサシステムの制御方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、複数のプロセッサを接続して高速に処理を行なう並列処理技術に関し、特に、消費電力を削減することが可能なマルチプロセッサシステムおよびその制御方法に関する。

【0002】

【従来の技術】

近年、マルチメディア処理や高精細画像処理などのさまざまな分野で、プロセッサ性能の向上に対する要求が高まっている。しかし、現在のLSI (Large Scale Integrated circuit) 製造技術ではデバイスの高速化にも限度がある。そこで、並列処理が注目を浴びており、マルチプロセッサシステムに関する研究、開発が盛んに行なわれている。

【0003】

一般に、複数のプロセッサが並列処理で1つの処理を行なう場合、できるだけ個々のプロセッサに均等な負荷分散を行なって、システム全体の性能を最大限に引き出す必要がある。しかし、実際のアプリケーション設計において、開発を容

易にしたり、システムコストの削減を考慮したりすると、必ずしも均等な負荷分散ができる訳ではない。したがって、一方のプロセッサが他方のプロセッサの処理完了を待たなければならない状況が多々発生する。

【0004】

一方、携帯機器に搭載された電池の寿命の向上や、近年盛んに唱えられている環境への配慮などのために、システムの低消費電力化が要求されている。このようなシステムの低消費電力化の一手法として、システムの稼動状況に応じてシステム全体や一部のクロック周波数を低減して、消費電力を抑えることを挙げることができる。

【0005】

図17は、従来のマルチプロセッサシステムにおける低消費電力化の一例を示す図である。このマルチプロセッサシステムは、プロセッサA101と、プロセッサB102と、プログラムAが格納されたメモリ103と、プログラムBが格納されたメモリ104と、プロセッサA101に入力されるクロックの供給／切断を制御するクロック供給制御部105とを含む。

【0006】

プロセッサA101およびプロセッサB102がそれぞれ、メモリ103に格納されたプログラムAおよびメモリ104に格納されたプログラムBを実行する場合において、プロセッサA101の処理時間の方がプロセッサB102の処理時間よりも短いとする。このとき、プロセッサA101がプログラムAの最後で、プロセッサB102の処理完了を待つことによって、2つのプロセッサの同期が取られていた。

【0007】

このようなシステムにおいて、プロセッサB102が処理を完了するまで、プロセッサA101が何も実行する必要がない場合、クロック供給制御部105にクロック制御指令を発行し、プロセッサA101に供給されるクロックを停止して、プロセッサA101の消費電力を削減していた。

【0008】

【発明が解決しようとする課題】

しかし、上述したマルチプロセッサシステムにおいて、プロセッサA101が処理を完了してから、プロセッサB102が処理を完了するまでの間、プロセッサA101のクロックを停止して消費電力を削減することは可能であるが、さらなる消費電力の削減を行なうことができないという問題点があった。

【0009】

本発明は、上記問題点を解決するためになされたものであり、第1の目的は、消費電力を大きく削減することが可能なマルチプロセッサシステムおよびその制御方法を提供することである。

【0010】

第2の目的は、消費電力を削減する際の処理を効率的に行なうことが可能なマルチプロセッサシステムおよびその制御方法を提供することである。

【0011】

【課題を解決するための手段】

請求項1に記載のマルチプロセッサシステムは、複数のプロセッサが並列に処理を行なった後に、同期を取るマルチプロセッサシステムであって、第1のプロセッサと、第1のプロセッサと異なる第2のプロセッサと、第1のプロセッサの処理時間と第2のプロセッサの処理時間との比率に応じて、第1のプロセッサに入力されるクロックの周波数を制御するためのクロック周波数制御手段とを含む。

【0012】

クロック周波数制御手段は、第1のプロセッサの処理時間と第2のプロセッサの処理時間との比率に応じて、第1のプロセッサに入力されるクロックの周波数を制御するので、第1のプロセッサに入力されるクロックの周波数を最適にすることができる、第1のプロセッサの消費電力を削減することが可能となる。

【0013】

請求項2に記載のマルチプロセッサシステムは、請求項1記載のマルチプロセッサシステムであって、さらに第1のプロセッサに入力されるクロックの周波数に応じて、第1のプロセッサに供給される電源電圧を制御するための電圧制御手段を含む。

【0014】

第1のプロセッサに入力されるクロックの周波数の低減に応じて、電源電圧を下げることができるため、さらに第1のプロセッサの消費電力を削減することが可能となる。

【0015】

請求項3に記載のマルチプロセッサシステムは、請求項1または2記載のマルチプロセッサシステムであって、さらに第1のプロセッサの処理時間見積値と第2のプロセッサの処理時間見積値とが予め格納される格納手段と、格納手段に格納された第1のプロセッサの処理時間見積値と第2のプロセッサの処理時間見積値との比率に応じてクロック周波数制御手段を制御して、第1のプロセッサに入力されるクロックの周波数を変更するための周波数決定手段とを含む。

【0016】

周波数決定手段は、第1のプロセッサの処理時間見積値と第2のプロセッサの処理時間見積値との比率に応じて、第1のプロセッサに入力されるクロックの周波数を変更するので、クロック周波数を変更するための処理を簡略化することが可能となる。

【0017】

請求項4に記載のマルチプロセッサシステムは、請求項1または2記載のマルチプロセッサシステムであって、さらに第1のプロセッサの処理時間見積値と第2のプロセッサの処理時間見積値との比率が予め格納される格納手段を含み、格納手段に格納された比率に応じてクロック周波数制御手段を制御して、第1のプロセッサに入力されるクロックの周波数を変更するための周波数決定手段とを含む。

【0018】

周波数決定手段は、格納手段に格納された比率に応じて、第1のプロセッサに入力されるクロックの周波数を変更するので、クロック周波数を変更するための処理を簡略化することが可能となる。

【0019】

請求項5に記載のマルチプロセッサシステムは、請求項1または2記載のマル

チプロセッサシステムであって、さらに前回の処理において、第1のプロセッサの処理が第2のプロセッサの処理よりも先に完了した場合には、クロック周波数制御手段に第1のプロセッサに入力されるクロックの周波数を低くさせ、前回の処理において、第2のプロセッサの処理が第1のプロセッサの処理よりも先に完了した場合には、第1のプロセッサに入力されるクロックの周波数を高くさせるための周波数決定手段を含む。

【0020】

周波数決定手段は、前回の第1のプロセッサおよび第2のプロセッサの処理状況に応じて、第1のプロセッサに入力されるクロックの周波数を変更するので、プロセッサの処理時間が実行毎に異なるような場合であっても、最適なクロック周波数を決定することが可能となる。

【0021】

請求項6に記載のマルチプロセッサシステムは、請求項1または2記載のマルチプロセッサシステムであって、さらに第1のプロセッサの処理と第2のプロセッサの処理とのいずれが先に完了するかを予測するための予測手段と、予測手段による予測結果に応じてクロック周波数制御手段を制御して、第1のプロセッサに入力されるクロックの周波数を変更するための周波数決定手段とを含む。

【0022】

周波数決定手段は、予測手段による予測結果に応じて、第1のプロセッサに入力されるクロックの周波数を変更するので、最適なクロック周波数を決定することが可能となる。

【0023】

請求項7に記載のマルチプロセッサシステムは、請求項6記載のマルチプロセッサシステムであって、予測手段は、前回の処理において、第1のプロセッサの処理が第2のプロセッサの処理よりも先に完了した場合には、第1のプロセッサの方が先に処理を完了すると予測し、前回の処理において、第2のプロセッサの処理が第1のプロセッサの処理よりも先に完了した場合には、第2のプロセッサの方が先に処理を完了すると予測する。

【0024】

したがって、第1のプロセッサが早く処理を完了したら、クロック周波数を低くしても、やはり第1のプロセッサが早く処理を完了する傾向にあるという戦略を表わすことができ、最適なクロック周波数を決定することが可能となる。

【0025】

請求項8に記載のマルチプロセッサシステムは、請求項6記載のマルチプロセッサシステムであって、予測手段は、前回の処理において、第1のプロセッサの処理が第2のプロセッサの処理よりも先に完了した場合には、第2のプロセッサの方が先に処理を完了すると予測し、前回の処理において、第2のプロセッサの処理が第1のプロセッサの処理よりも先に完了した場合には、第1のプロセッサの方が先に処理を完了すると予測する。

【0026】

したがって、第1のプロセッサが早く処理を完了したら、クロック周波数を低くすることになるので、次は第2のプロセッサが早く処理を完了する傾向にあるという戦略を表わすことができ、最適なクロック周波数を決定することが可能となる。

【0027】

請求項9に記載のマルチプロセッサシステムは、請求項1または2記載のマルチプロセッサシステムであって、さらに第1のプロセッサの処理時間を計測するための第1の計測手段と、第2のプロセッサの処理時間を計測するための第2の計測手段と、第1の計測手段による計測結果および第2の計測手段による計測結果に応じてクロック周波数制御手段を制御して、第1のプロセッサに入力されるクロックの周波数を変更するための周波数決定手段とを含む。

【0028】

周波数決定手段は、第1の計測手段による計測結果および第2の計測手段による計測結果に応じてクロック周波数制御手段を制御するので、最適なクロック周波数を決定することが可能となる。

【0029】

請求項10に記載のマルチプロセッサシステムの制御方法は、複数のプロセッサが並列に処理を行なった後に、同期を取るマルチプロセッサシステムの制御方

法であって、第1のプロセッサの処理時間と第2のプロセッサの処理時間との比率を算出するステップと、算出された比率に応じて、第1のプロセッサに入力されるクロックの周波数を制御するステップとを含む。

【0030】

第1のプロセッサの処理時間と第2のプロセッサの処理時間との比率に応じて、第1のプロセッサに入力されるクロックの周波数を制御するので、第1のプロセッサに入力されるクロックの周波数を最適にすことができ、第1のプロセッサの消費電力を削減することが可能となる。

【0031】

請求項11に記載のマルチプロセッサシステムの制御方法は、請求項10記載のマルチプロセッサの制御方法であって、さらに第1のプロセッサに入力されるクロックの周波数に応じて、第1のプロセッサに供給される電源電圧を制御するステップを含む。

【0032】

第1のプロセッサに入力されるクロックの周波数の低減に応じて、電源電圧を下げるができるため、さらに第1のプロセッサの消費電力を削減することが可能となる。

【0033】

【発明の実施の形態】

(実施の形態1)

図1は、本発明の実施の形態1におけるマルチプロセッサシステムの概略構成を示すブロック図である。このマルチプロセッサシステムは、プロセッサA11と、プロセッサB12と、プログラムAおよびプログラムBの処理時間見積値Taが格納されたメモリ13と、プログラムBおよびプログラムBの処理時間見積値Tbが格納されたメモリ14と、プロセッサA11に入力されるクロック周波数を制御するクロック周波数制御部15と、プロセッサA11に電源電圧を供給する可変電源16とを含む。

【0034】

プロセッサA11が、メモリ13に格納されたプログラムAを実行し、プロセ

ツサB12が、メモリ14に格納されたプログラムBを実行することによって、並列に処理が行なわれる。このシステムは、プロセッサA11とプロセッサB12との間で情報を通信するための通信機能を有しており、たとえば、割込み、メモリマップドI/O (Input/Output)、共有メモリ等によって実現される。この通信機能は、プロセッサA11がプロセッサB12に対してプログラムBの実行を起動する際、およびプロセッサB12がプログラムBの処理を完了したことをプロセッサA11に対して通知する際に使用される。

【0035】

プロセッサA11は、メモリ13に格納されたプログラムAの処理時間見積値T_aおよびメモリ14に格納されたプログラムBの処理時間見積値T_bを参照して、プロセッサA11に入力されるクロック周波数を決定し、クロック周波数制御部15にクロック制御指令を発行する。プログラムAの処理時間見積値T_aおよびプログラムBの処理時間見積値T_bは、たとえば、プログラムA、Bを作成するプログラマによって見積もられた値であり、予めメモリ13およびメモリ14に格納されている。なお、クロック周波数の決定方法の詳細は後述する。

【0036】

図2は、クロック周波数制御部15の詳細を説明するための図である。このクロック周波数制御部15は、発振回路17と、発振回路17から出力されるクロックを分周する複数の分周器18-1～18-mと、発振回路17および複数の分周器18-1～18-mの出力のいずれかを選択して出力するセレクタ19とを含む。

【0037】

発振回路17は、プロセッサA11に入力されるクロックの最高周波数以上のクロックを出力する。分周器18-1～18-mは、それぞれ発振回路17から出力されるクロックを $1/2^i$ ($i = 1, 2, \dots, m$) に分周する。セレクタ19は、プロセッサA11から出力されるクロック制御指令（周波数選択信号）に応じて、発振回路17および分周器18-1～18-mからの出力を選択して出力する。

【0038】

図3は、本実施の形態におけるプロセッサA11の処理手順を説明するためのフローチャートである。まず、プロセッサA11は、プロセッサB12に対して起動要求を出力して、プロセッサB12にプログラムBの実行を要求する(S11)。そして、プロセッサA11は、メモリ13に格納されたプログラムAの処理時間見積値Taおよびメモリ14に格納されたプログラムBの処理時間見積値Tbを参照し(S12)、見積値Tbが見積値Taよりも大きいか否かを判定する(S13)。見積値Tbが見積値Ta以下であれば(S13, No)、プロセッサA11のクロック周波数を変更せずに、ステップS16へ処理が進む。

【0039】

また、見積値Tbが見積値Taよりも大きければ(S13, Yes)、プロセッサA11は、プログラムAの処理時間見積値TaおよびプログラムBの処理時間見積値Tbから、次式に示す範囲内において、プロセッサA11に新たに入力されるクロック周波数を決定して設定する(S14)。なお、fは元のクロック周波数を、f'は新たなクロック周波数を示している。

【0040】

$$f \times T_a / T_b \leqq f' < f \quad \dots (1)$$

たとえば、 $T_b / 4 < T_a \leqq T_b / 2$ の場合には、1/2分周器18-1から出力されるクロックが選択されるように、プロセッサA11がセレクタ19にクロック制御指令を出力する。また、 $T_b / 8 < T_a \leqq T_b / 4$ の場合には、1/4分周器18-2から出力されるクロックが選択されるように、プロセッサA11がセレクタ19にクロック制御指令を出力する。

【0041】

次に、プロセッサA11は、可変電源16を制御してプロセッサA11に供給される電源電圧を下げ(S15)、プログラムAの実行を開始する(S16)。プロセッサA11は、プログラムAの実行を完了すると、上述した通信機能を用いてプロセッサB12から処理完了の通知があるか否かを判定する(S17)。プロセッサB12から処理完了の通知がなければ(S17, No)、処理完了の通知があるまで待機する。

【0042】

また、プロセッサB12から処理完了の通知があれば（S17, Yes）、プロセッサA11は、可変電源16を制御してプロセッサA11に供給される電源電圧を元に戻す。そして、プロセッサA11はクロック周波数制御部15に対してクロック制御指令を発行して、クロック周波数を元のクロック周波数に戻して（S18）、処理を終了する。

【0043】

なお、このようにクロック周波数が低くなると電源電圧を低くすることができるのは、クロック周波数が低くなると各信号の遷移時間（遅延時間）が多少大きくなってしまって回路動作に支障が生じないためである。クロック周波数と許容できる電圧値との関係は予め実験等によって定めることができる。上述した可変電源16の制御は、予めクロック周波数に対応した電源電圧値をテーブルに格納しておき、プロセッサA11がそのテーブルを参照することによって行なわれる。

【0044】

図4は、図3に示すプロセッサA11の処理手順を、C言語風の疑似プログラムによって表わしたものである。このプログラムにおいて、“B_is_done”はプロセッサB12によるプログラムBの実行状態を示しており、“0”的ときはプログラムBの実行中であることを示し、“1”的ときはプログラムBの実行が完了していることを示している。なお、各ステップの処理内容は、コメント文で詳細に説明されているので、ここでの説明は行なわない。

【0045】

図4に示すプログラムにおいては、プロセッサB12からのプログラムBの完了通知は、割込みによって行なわれている。割込み以外によって通信機能を実現する場合には、上述したメモリマップドI/Oまたは共有メモリをポーリングすることによって、プロセッサB12からのプログラムBの完了通知を得ることができる。

【0046】

以上説明したように、本実施の形態におけるマルチプロセッサシステムによれば、プロセッサA11とプロセッサB12とが並列に処理を実行する際、プロセッサA11がプロセッサB12よりも先に処理が完了する場合には、プロセッサ

A11を低速動作させるようにした。したがって、CMOS (Complementary Metal Oxide Semiconductor) 論理回路の消費電力がクロック周波数に比例することから、プロセッサA11の消費電力を削減することが可能となった。

【0047】

また、プロセッサA11を低速動作させることにより、プロセッサA11の電源電圧を下げることができるようになった。したがって、CMOS論理回路の消費電力が電源電圧の2乗に比例することから、プロセッサA11の消費電力をさらに削減することが可能となった。

【0048】

(実施の形態2)

図5は、本発明の実施の形態2におけるマルチプロセッサシステムの概略構成を示すブロック図である。本実施の形態におけるマルチプロセッサシステムは、図1に示す実施の形態1におけるマルチプロセッサシステムと比較して、メモリ13および14のそれぞれにプログラムAの処理時間見積値TaおよびプログラムBの処理時間見積値Tbが予め格納される代わりに、メモリ13にプログラムAとBとの処理時間比率R (=Ta/Tb) が予め格納されている点のみが異なる。したがって、重複する構成および機能の詳細な説明は繰返さない。なお、処理時間比率Rは、たとえば、プログラムA、Bを作成するプログラマによって見積もられた値であり、予めメモリ13に格納されている。

【0049】

図6は、本実施の形態におけるプロセッサA11の処理手順を説明するためのフローチャートである。本実施の形態におけるプロセッサA11の処理手順は、図3に示す実施の形態1におけるプロセッサA11の処理手順と比較して、ステップS12およびS13がステップS22に置換されている点のみが異なる。したがって、重複する処理手順の詳細な説明は繰返さない。

【0050】

プロセッサA11は、プロセッサB12に起動要求を出力し (S11) 、メモリ13に格納されたプログラムAとBとの処理時間比率Rを参照する (S22) 。そして、プロセッサA11は、次式に示す範囲内において、プロセッサA11

に新たに入力されるクロック周波数を決定して設定する（S14）。なお、 f は元のクロック周波数を、 f' は新たなクロック周波数を示している。

【0051】

$$R \leq f' / f < 1 \quad \cdots (2)$$

たとえば、 $1/4 < R \leq 1/2$ の場合には、 $1/2$ 分周器 18-1 から出力されるクロックが選択されるように、プロセッサ A11 がセレクタ 19 にクロック制御指令を出力する。また、 $1/8 < R \leq 1/4$ の場合には、 $1/4$ 分周器 18-2 から出力されるクロックが選択されるように、プロセッサ A11 がセレクタ 19 にクロック制御指令を出力する。

【0052】

以上説明したように、本実施の形態におけるマルチプロセッサシステムによれば、実施の形態 1 において説明した効果に加えて、新たなクロック周波数を決定するための処理が簡略化されて、プロセッサ A11 の処理負担を軽減することが可能となった。なお、本実施の形態におけるマルチプロセッサシステムは、プログラム A がプログラム B によって実現される機能を呼出す場合であって、常にプログラム A のみがプログラム B を呼出すようにプログラミングされている場合に有効である。

【0053】

（実施の形態 3）

図 7 は、本発明の実施の形態 3 におけるマルチプロセッサシステムの概略構成を示すブロック図である。本実施の形態におけるマルチプロセッサシステムは、図 1 に示す実施の形態 1 におけるマルチプロセッサシステムと比較して、クロック周波数制御部 15 からのクロック信号および可変電源 16 からの電源電圧がプロセッサ B12 に供給される点のみが異なる。したがって、重複する構成および機能の詳細な説明は繰返さない。

【0054】

図 8 は、本実施の形態におけるプロセッサ A11 の処理手順を説明するためのフローチャートである。本実施の形態におけるプロセッサ A11 の処理手順は、図 3 に示す実施の形態 1 におけるプロセッサ A11 の処理手順と比較して、ステ

ップS14およびS15がステップS34およびS35に置換されている点のみが異なる。したがって、重複する処理手順の詳細な説明は繰返さない。

【0055】

ステップS13において、見積値Tbが見積値Taよりも大きければ(S13, Yes)、プロセッサA11は、可変電源16を制御してプロセッサA11に供給される電源電圧を上げる(S34)。

【0056】

次に、プロセッサA11は、プログラムAの処理時間見積値TaおよびプログラムBの処理時間見積値Tbから、次式に示す範囲内において、プロセッサB12に新たに入力されるクロック周波数を決定して設定する(S35)。なお、fは元のクロック周波数を、f'は新たなクロック周波数を示している。また、元のクロック周波数fは、1/16分周器18-4からの出力が選択されている場合のクロック周波数であるとする。

【0057】

$$f < f' \leq f \times T_b / T_a \quad \cdots (3)$$

たとえば、 $2T_a \leq T_b < 4T_a$ の場合には、1/8分周器18-3から出力されるクロック(元のクロック周波数fの2倍)が選択されるように、プロセッサA11がセレクタ19にクロック制御指令を出力する。また、 $4T_a \leq T_b < 8T_a$ の場合には、1/4分周器18-2から出力されるクロック(元のクロック周波数fの4倍)が選択されるように、プロセッサA11がセレクタ19にクロック制御指令を出力する。そして、ステップS16以降の処理を行なう。

【0058】

以上説明したように、本実施の形態におけるマルチプロセッサシステムによれば、プロセッサA11とプロセッサB12とが並列に処理を実行する際、プロセッサA11がプロセッサB12よりも先に処理が完了する場合にのみ、プロセッサB12を高速動作させるようにした。したがって、CMOS論理回路の消費電力がクロック周波数に比例することから、プロセッサB12の通常の処理状態(高速動作させない状態)における消費電力を削減することが可能となった。

【0059】

また、プロセッサB12の通常の処理状態において低速動作させることにより、プロセッサB12の電源電圧を下げることができるようになった。したがって、CMOS論理回路の消費電力が電源電圧の2乗に比例することから、プロセッサB12の通常状態における消費電力をさらに削減することが可能となった。なお、本実施の形態におけるマルチプロセッサシステムは、予めマルチプロセッサシステム全体を低速動作させておき、必要に応じて処理の一部でプロセッサを高速に動作させるようなシステムに有効である。

【0060】

(実施の形態4)

実施の形態1～3のマルチプロセッサシステムにおいては、プログラマが予め、プログラムAの処理時間見積値TaとプログラムBの処理時間見積値Tbとを決めておき、その値に応じてクロック周波数や電源電圧を変更して低消費電力化を実現していた。しかし、プログラムAの処理時間やプログラムBの処理時間が実行毎に異なるような場合には、実施の形態1～3のマルチプロセッサシステムを構築することができない。

【0061】

本発明の実施の形態4におけるマルチプロセッサシステムは、プログラムAとプログラムBとが、プロセッサA11とプロセッサB12とによって繰返し並列に実行されるような場合に、前回の処理においてプロセッサA11がプロセッサB12からのプログラムBの処理完了通知を待ったか否かによって、クロック周波数を変更するものである。

【0062】

図9は、本発明の実施の形態4におけるマルチプロセッサシステムの概略構成を示すブロック図である。本実施の形態におけるマルチプロセッサシステムは、図1に示す実施の形態1におけるマルチプロセッサシステムと比較して、メモリ13および14のそれぞれにプログラムAの処理時間見積値TaおよびプログラムBの処理時間見積値Tbが格納されていない点のみが異なる。したがって、重複する構成および機能の詳細な説明は繰返さない。なお、図2に示す実施の形態1におけるクロック制御部15は、発振回路17および複数の分周器18-1～

18-mから出力されるクロックを切替えて、 $1/2^i$ 分周されたクロックを出力するものであったが、本実施の形態におけるクロック制御部はさらに細かく段階的にクロックを切替えることができるものとする。

【0063】

図10は、本発明の実施の形態4におけるプロセッサA11の処理手順を説明するためのフローチャートである。まず、プロセッサA11は、プロセッサB12に対して起動要求を出力して、プロセッサB12にプログラムBの実行を要求する(S41)。そして、プロセッサA11は、前回の処理に対応する情報I(n-1)を参照し(S42)、情報I(n-1)と予め定められたしきい値WAIT_LIMITとを比較して、前回のプログラムAの処理においてプロセッサB12からの完了通知を待ったか否かを判定する(S43)。

【0064】

プロセッサA11が、プロセッサB12からの完了通知を待った場合、すなわち情報I(n-1)がしきい値WAIT_LIMITよりも大きい場合には(S43, Yes)、クロック周波数制御部15を制御して、クロック周波数を1段だけ遅くする(S44)。また、プロセッサA11が、プロセッサB12からの完了通知を待たなかった場合、すなわち情報I(n-1)がしきい値WAIT_LIMIT以下の場合には(S43, No)、プロセッサA11がプログラムAの処理を完了する前に、プロセッサB12が処理を完了していたか否かを判定する(S45)。

【0065】

プロセッサA11がプログラムAの処理を完了する前に、プロセッサB12が処理を完了していた場合、すなわち情報I(n-1)が“0”的場合には(S45, Yes)、クロック制御部15を制御して、クロック周波数を1段だけ速くする(S46)。また、プロセッサA11がプログラムAの処理を完了するのと同じくらいに、プロセッサB12が処理を完了していた場合、すなわち情報I(n-1)が“0”でない場合には(S45, No)、クロック周波数を切替えずに、ステップS47へ進む。

【0066】

次に、プロセッサA11はプロセッサA11内に設けられたレジスタwait_countに“0”を代入し(S47)、プログラムAの実行を開始する(S48)。プロセッサA11は、プログラムAの実行を完了すると、上述した通信機能を用いてプロセッサB12から処理完了の通知があるか否かを判定する(S49)。プロセッサB12から処理完了の通知がなければ(S49, No)、wait_countの値をカウントアップし(S50)、ステップS49へ戻る。また、プロセッサB12から処理完了の通知があれば(S49, Yes)、情報I(n)にwait_countの値を代入して、情報I(n)を決定し(S51)、処理を終了する。なお、nはプロセッサA11がプログラムAの処理を行なった回数を示しており、この情報I(n)の値は次回の処理においてステップS42で参照される。

【0067】

図11は、図10に示すプロセッサA11の処理手順を、C言語風の疑似プログラムによって表わしたものである。このプログラムにおいて、“B_is_done”はプロセッサB12によるプログラムBの実行状態を示しており、“0”のときはプログラムBの実行中であることを示し、“1”のときはプログラムBの実行が完了していることを示している。なお、各ステップの処理内容は、コメント文で詳細に説明されているので、ここでの説明は行なわない。

【0068】

以上説明したように、本実施の形態におけるマルチプロセッサシステムによれば、プロセッサA11とプロセッサB12とが並列に処理を実行する際、プロセッサA11の処理とプロセッサB12の処理とのいずれが先に完了するかによって、プロセッサA11のクロック周波数を切替えてクロック周波数が最適となるようにした。したがって、CMOS論理回路の消費電力がクロック周波数に比例することから、プロセッサA11の消費電力を削減することが可能となった。

【0069】

また、実施の形態1～3におけるマルチプロセッサシステムと同様に、プロセッサA11のクロック周波数を切替えながら、プロセッサA11の電源電圧を細かく制御するようにすれば、CMOS論理回路の消費電力が電源電圧の2乗に比

例することから、プロセッサA11の消費電力をさらに削減することが可能となる。

【0070】

(実施の形態5)

実施の形態4のマルチプロセッサシステムにおいては、前回の処理においてプロセッサA11がプロセッサB12からのプログラムBの処理完了通知を待ったか否かによって、クロック周波数を変更するものであった。本発明の実施の形態5におけるマルチプロセッサシステムは、過去の複数回の処理においてプロセッサA11がプロセッサB12からのプログラムBの処理完了通知を待ったか否かによって、次の処理においてプロセッサA11とプロセッサB12とのいずれが早く処理を完了するかを予測して、クロックを制御するものである。

【0071】

本発明の実施の形態におけるマルチプロセッサシステムの概略構成は、図4に示す実施の形態4におけるマルチプロセッサシステムの概略構成と同じである。したがって、重複する構成および機能の詳細な説明は繰返さない。

【0072】

図12は、本実施の形態におけるプロセッサA11の処理手順を説明するためのフローチャートである。まず、プロセッサA11は、プロセッサB12に対して起動要求を出力して、プロセッサB12にプログラムBの実行を要求する(S61)。そして、プロセッサA11は、プロセッサA11内に設けられた状態レジスタの値によって現在の状態を参照し(S62)、プロセッサA11(プログラムA)の方がプロセッサB12(プログラムB)よりも早く処理を完了すると予測したか否かを判定する(S63)。なお、状態レジスタは、後述する図14または図15に示す状態の番号が保持されるレジスタであり、この状態レジスタの値を更新することによって状態の遷移が行なわれる。

【0073】

プロセッサA11が、プロセッサA11の方がプロセッサB12よりも早く処理を完了すると予測した場合には(S63, Yes)、クロック周波数制御部15を制御して、クロック周波数を1段だけ遅くする(S64)。また、プロセッ

サA11が、プロセッサA11の方がプロセッサB12よりも早く処理を完了すると予測しなかった場合には（S63, No）、プロセッサB12の方がプロセッサA11よりも早く処理を完了すると予測したか否かを判定する（S65）。

【0074】

プロセッサB12の方がプロセッサA11よりも早く処理を完了すると予測した場合には（S65, Yes）、クロック制御部15を制御して、クロック周波数を1段だけ速くする（S66）。また、プロセッサB12の方がプロセッサA11よりも早く処理を完了すると予測しなかった場合には（S65, No）、クロック周波数を切替えずに、ステップS67へ進む。

【0075】

次に、プロセッサA11はプロセッサA11内に設けられたレジスタwait_countに“0”を代入し（S67）、プログラムAの実行を開始する（S68）。プロセッサA11は、プログラムAの実行を完了すると、上述した通信機能を用いてプロセッサB12から処理完了の通知があるか否かを判定する（S69）。プロセッサB12から処理完了の通知がなければ（S69, No）、wait_countの値をカウントアップし（S70）、ステップS69へ戻る。また、プロセッサB12から処理完了の通知があれば（S69, Yes）、wait_countの値により状態を更新し（S71）、処理を終了する。なお、この状態の更新については後述する。

【0076】

図13は、図12に示すプロセッサA11の処理手順を、C言語風の疑似プログラムによって表わしたものである。このプログラムにおいて、“B_is_done”はプロセッサB12によるプログラムBの実行状態を示しており、“0”のときはプログラムBの実行中であることを示し、“1”のときはプログラムBの実行が完了していることを示している。また、“wait_count”は、プロセッサA11がどれだけプロセッサB12の処理完了を待ったかを示している。この“wait_count”的値によって、後述する状態の遷移が決定される。なお、各ステップの処理内容は、コメント文で詳細に説明されているので、ここでの説明は行なわない。

【0077】

図14は、次の処理においてプロセッサA11とプロセッサB12とのいずれが早く処理を完了するかを予測するための状態遷移図である。“現在の状態から予測値を得る”および“状態を更新する”の2つの手続は、発見的手法により様々な方法が考えられる。図14は、その一例を示しており、楕円は状態を表わし、矢印は状態の更新を表わしている。この状態遷移図は、プロセッサA11（プログラムA）とプロセッサB12（プログラムB）とのいずれが早く処理を完了したかによって、状態をどのように遷移させるかを規定しており、それぞれの状態には予測値が割当てられている。

【0078】

たとえば、現在の状態が“状態1”にあるとき、“現在の状態から予測値を取出す”という操作によって、“状態1”に割当てられている“プログラムAが早く処理を完了すると予測”という予測値が得られる。また、現在の状態が“状態1”にあるとき、プログラムBが早く完了した場合には“状態を更新する”という操作によって、“状態1”から“状態2”へ状態が更新される。

【0079】

図14に示す状態遷移図に準拠して図12のステップS71の処理が行なわれる場合、あるしきい値WAIT_LIMIT (≥ 0) が予め定められており、wait_countの値がWAIT_LIMITを超えたときに“プログラムAが早く完了した”と判断し、そうでないならば“プログラムBが早く完了した”と判断して、現在の状態レジスタの値を参照して図14に示す状態遷移が生じるように状態を更新する。

【0080】

図14に示す状態遷移図においては、“プログラムAが早く処理を完了したら、クロック周波数を低くしても、やはりプログラムAが早く処理を完了する傾向にある”という戦略を表現している。これに対して、“プログラムAが早く処理を完了したら、クロック周波数を低くすることになるので、次はプログラムBが早く処理を完了する傾向がある”という戦略が考えられる。図15は、この戦略に基づく状態遷移図を示しており、プログラムAの処理時間とプログラムBの処

理時間とがちょうどつりあう状態も考慮している。

【0081】

図15に示す状態遷移図に準拠して図12のステップS71の処理が行なわれる場合、ある2つのしきい値WAIT_LIMIT1、WAIT_LIMIT2（ただし、 $0 \leq \text{WAIT_LIMIT1} < \text{WAIT_LIMIT2}$ ）が予め定められており、wait_countの値がWAIT_LIMIT2を超える場合には“プログラムAが早く完了した”と判断し、wait_countの値がWAIT_LIMIT1を超えてWAIT_LIMIT2以下である場合には“プログラムA、Bはほぼ同時に処理を完了した”と判断し、さらにwait_countの値がWAIT_LIMIT1以下である場合には“プログラムBが早く完了した”と判断し、現在の状態レジスタの値を参照して図15に示す状態遷移が生じるように状態を更新する。

【0082】

以上説明したように、本実施の形態におけるマルチプロセッサシステムによれば、プロセッサA11とプロセッサB12とが並列に処理を実行する際、プロセッサA11の処理とプロセッサB12の処理とのいずれが先に完了するかを予測して、プロセッサA11のクロック周波数を切替えてクロック周波数が最適となるようにした。したがって、CMOS論理回路の消費電力がクロック周波数に比例することから、プロセッサA11の消費電力を削減することが可能となった。

【0083】

また、実施の形態1～3におけるマルチプロセッサシステムと同様に、プロセッサA11のクロック周波数を切替えながら、プロセッサA11の電源電圧を細かく制御するようにすれば、CMOS論理回路の消費電力が電源電圧の2乗に比例することから、プロセッサA11の消費電力をさらに削減することが可能となる。

【0084】

（実施の形態6）

図16は、本発明の実施の形態6におけるマルチプロセッサシステムの概略構成を示すブロック図である。本実施の形態におけるマルチプロセッサシステムは

、図1に示す実施の形態1におけるマルチプロセッサシステムと比較して、プロセッサA11によるプログラムAの処理時間を計測するカウンタ21と、プロセッサB12によるプログラムBの処理時間を計測するカウンタ22とが追加された点のみが異なる。したがって、重複する構成および機能の詳細な説明は繰返さない。

【0085】

カウンタ21および22は、共通のクロック（カウンタ用クロック）をカウントすることによってカウントアップする。このカウント用クロックには、プロセッサA11に供給されるクロックおよびプロセッサB12に供給されるクロックよりも周波数が小さいものが使用されるのが好ましい。なぜなら、低消費電力化のためである。

【0086】

カウンタ21は、プロセッサA11がプログラムAの実行を開始するときにリセットされ、プログラムAの実行中にカウントアップする。そして、プロセッサA11がプログラムAの実行を終了するときに、カウンタ21のカウント値が処理時間見積値T_aとしてメモリ13に転送される。また、カウンタ22は、プロセッサB12がプログラムBの実行を開始するときにリセットされ、プログラムBの実行中にカウントアップする。そして、プロセッサB12がプログラムBの実行を終了するときに、カウンタ22のカウント値が処理時間見積値T_bとしてメモリ14に転送される。

【0087】

プロセッサA11は、次回のプロセッサA11によるプログラムAの実行時において、メモリ13に格納されたプログラムAの処理時間見積値T_aおよびメモリ14に格納されたプログラムBの処理時間見積値T_bを参照して、プロセッサA11に入力されるクロック周波数を決定し、クロック周波数制御部15にクロック制御指令を発行する。

【0088】

本実施の形態におけるマルチプロセッサシステムの処理手順は、図3に示す実施の形態1におけるマルチプロセッサシステムの処理手順と同じであるので、詳

細な説明は繰返さない。

【0089】

以上説明したように、本実施の形態におけるマルチプロセッサシステムによれば、プロセッサA11とプロセッサB12とが並列に処理を実行する際、前回のプログラムAの処理時間の計測値とプログラムBの処理時間の計測値とを参照し、プロセッサA11がプロセッサB12よりも先に処理が完了する場合には、プロセッサA11を低速動作させるようにした。したがって、CMOS論理回路の消費電力がクロック周波数に比例することから、プロセッサA11の消費電力を削減することが可能となった。

【0090】

また、プロセッサA11を低速動作させることにより、プロセッサA11の電源電圧を下げる様にした。したがって、CMOS論理回路の消費電力が電源電圧の2乗に比例することから、プロセッサA11の消費電力をさらに削減することが可能となった。

【0091】

今回開示された実施の形態は、すべての点で例示であって制限的なものではないと考えられるべきである。本発明の範囲は上記した説明ではなくて特許請求の範囲によって示され、特許請求の範囲と均等の意味および範囲内でのすべての変更が含まれることが意図される。

【0092】

【発明の効果】

請求項1に記載のマルチプロセッサシステムによれば、クロック周波数制御手段が、第1のプロセッサの処理時間と第2のプロセッサの処理時間との比率に応じて、第1のプロセッサに入力されるクロックの周波数を制御するので、第1のプロセッサに入力されるクロックの周波数を最適にすることができ、第1のプロセッサの消費電力を削減することが可能となった。

【0093】

請求項2に記載のマルチプロセッサシステムによれば、第1のプロセッサに入力されるクロックの周波数の低減に応じて、電源電圧を下げるため

、さらに第1のプロセッサの消費電力を削減することが可能となった。

【0094】

請求項3に記載のマルチプロセッサシステムによれば、周波数決定手段が、第1のプロセッサの処理時間見積値と第2のプロセッサの処理時間見積値との比率に応じて、第1のプロセッサに入力されるクロックの周波数を変更するので、クロック周波数を変更するための処理を簡略化することが可能となった。

【0095】

請求項4に記載のマルチプロセッサシステムによれば、周波数決定手段が、格納手段に格納された比率に応じて、第1のプロセッサに入力されるクロックの周波数を変更するので、クロック周波数を変更するための処理を簡略化することが可能となった。

【0096】

請求項5に記載のマルチプロセッサシステムによれば、周波数決定手段が、前回の第1のプロセッサおよび第2のプロセッサの処理状況に応じて、第1のプロセッサに入力されるクロックの周波数を変更するので、プロセッサの処理時間が実行毎に異なるような場合であっても、最適なクロック周波数を決定することが可能となった。

【0097】

請求項6に記載のマルチプロセッサシステムによれば、周波数決定手段が、予測手段による予測結果に応じて、第1のプロセッサに入力されるクロックの周波数を変更するので、最適なクロック周波数を決定することが可能となった。

【0098】

請求項7に記載のマルチプロセッサシステムによれば、第1のプロセッサが早く処理を完了したら、クロック周波数を低くしても、やはり第1のプロセッサが早く処理を完了する傾向にあるという戦略を表わすことができ、最適なクロック周波数を決定することが可能となった。

【0099】

請求項8に記載のマルチプロセッサシステムによれば、第1のプロセッサが早く処理を完了したら、クロック周波数を低くすることになるので、次は第2のプ

ロセッサが早く処理を完了する傾向にあるという戦略を表わすことができ、最適なクロック周波数を決定することが可能となった。

【0100】

請求項9に記載のマルチプロセッサシステムによれば、周波数決定手段が、第1の計測手段による計測結果および第2の計測手段による計測結果に応じてクロック周波数制御手段を制御するので、最適なクロック周波数を決定することが可能となった。

【0101】

請求項10に記載のマルチプロセッサシステムの制御方法によれば、第1のプロセッサの処理時間と第2のプロセッサの処理時間との比率に応じて、第1のプロセッサに入力されるクロックの周波数を制御するので、第1のプロセッサに入力されるクロックの周波数を最適にすることができ、第1のプロセッサの消費電力を削減することが可能となった。

【0102】

請求項11に記載のマルチプロセッサシステムの制御方法によれば、第1のプロセッサに入力されるクロックの周波数の低減に応じて、電源電圧を下げることができるため、さらに第1のプロセッサの消費電力を削減することが可能となった。

【図面の簡単な説明】

【図1】 本発明の実施の形態1におけるマルチプロセッサシステムの概略構成を示すブロック図である。

【図2】 クロック周波数制御部15の詳細を説明するための図である。

【図3】 本発明の実施の形態1におけるプロセッサA11の処理手順を説明するためのフローチャートである。

【図4】 図3に示すプロセッサA11の処理手順をC言語風の疑似プログラムによって表わした図である。

【図5】 本発明の実施の形態2におけるマルチプロセッサシステムの概略構成を示すブロック図である。

【図6】 本発明の実施の形態2におけるプロセッサA11の処理手順を説

明するためのフローチャートである。

【図7】 本発明の実施の形態3におけるマルチプロセッサシステムの概略構成を示すブロック図である。

【図8】 本発明の実施の形態3におけるプロセッサA11の処理手順を説明するためのフローチャートである。

【図9】 本発明の実施の形態4におけるマルチプロセッサシステムの概略構成を示すブロック図である。

【図10】 本発明の実施の形態4におけるプロセッサA11の処理手順を説明するためのフローチャートである。

【図11】 図10に示すプロセッサA11の処理手順をC言語風の疑似プログラムによって表わした図である。

【図12】 本発明の実施の形態5におけるプロセッサA11の処理手順を説明するためのフローチャートである。

【図13】 図12に示すプロセッサA11の処理手順をC言語風の疑似プログラムによって表わした図である。

【図14】 プロセッサA11とプロセッサB12とのいづれが早く処理を完了するかを予測するための状態遷移の一例を示す図である。

【図15】 プロセッサA11とプロセッサB12とのいづれが早く処理を完了するかを予測するための状態遷移の他の一例を示す図である。

【図16】 本発明の実施の形態6におけるマルチプロセッサシステムの概略構成を示すブロック図である。

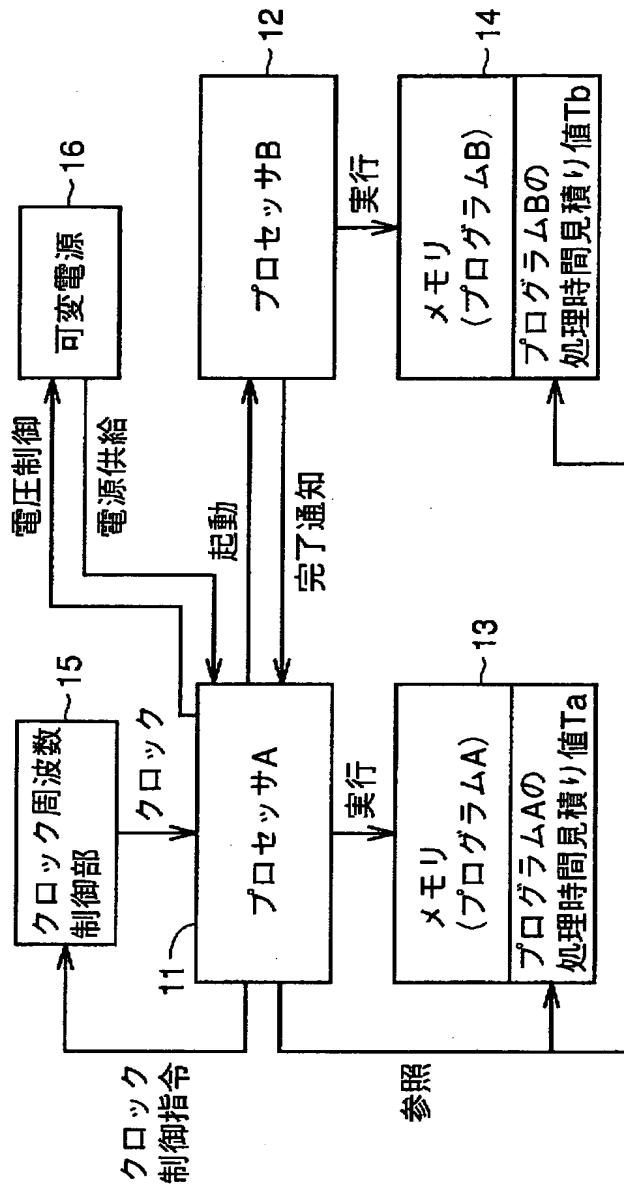
【図17】 従来のマルチプロセッサシステムの概略構成を示すブロック図である。

【符号の説明】

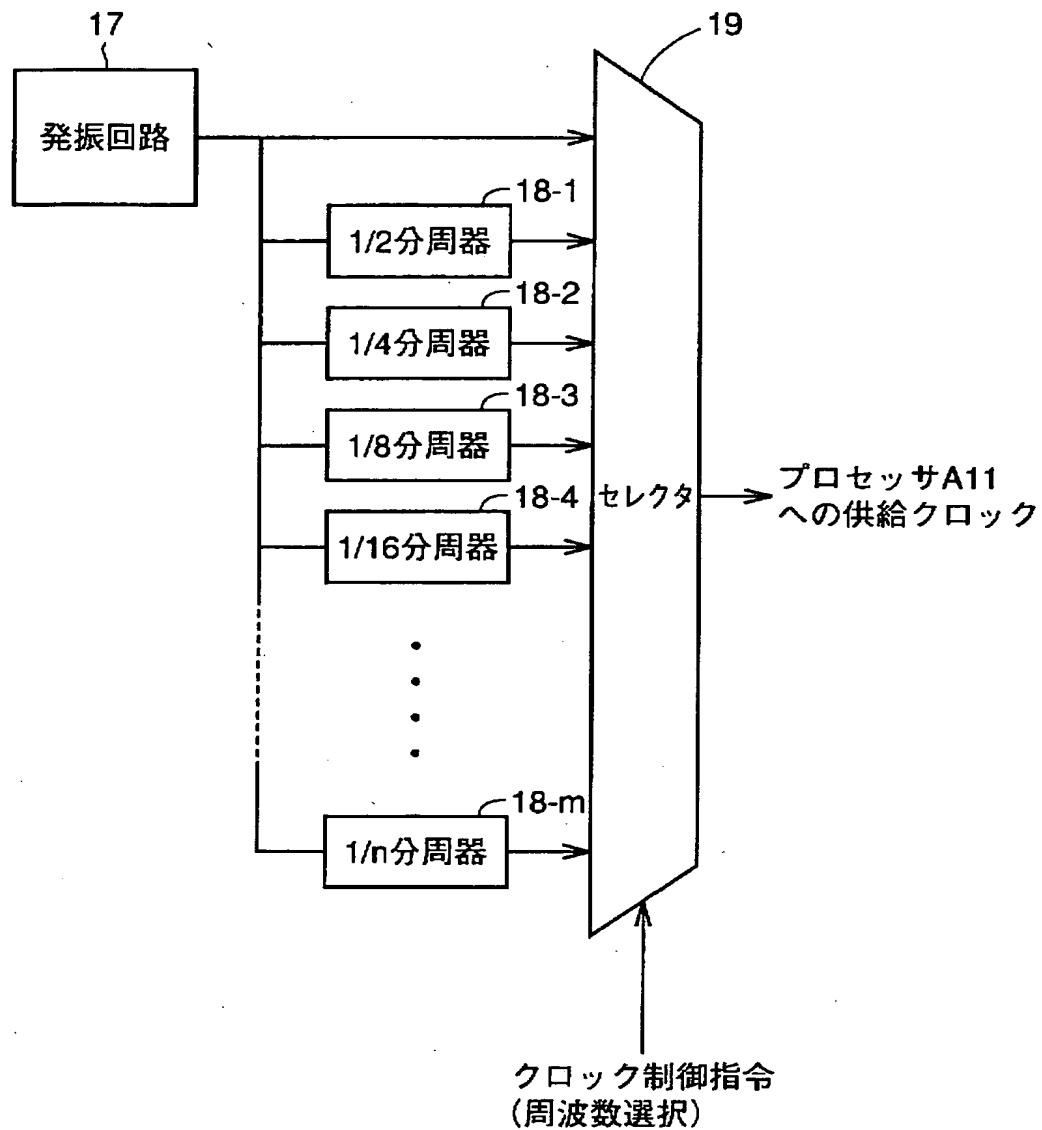
11 プロセッサA、12 プロセッサB、13, 14 メモリ、15 クロック周波数制御部、16 可変電源、17 発振回路、18-1～18-m 分周器、19 セレクタ、21 プログラムAの処理時間計測カウンタ、22 プログラムBの処理時間計測カウンタ。

【書類名】 図面

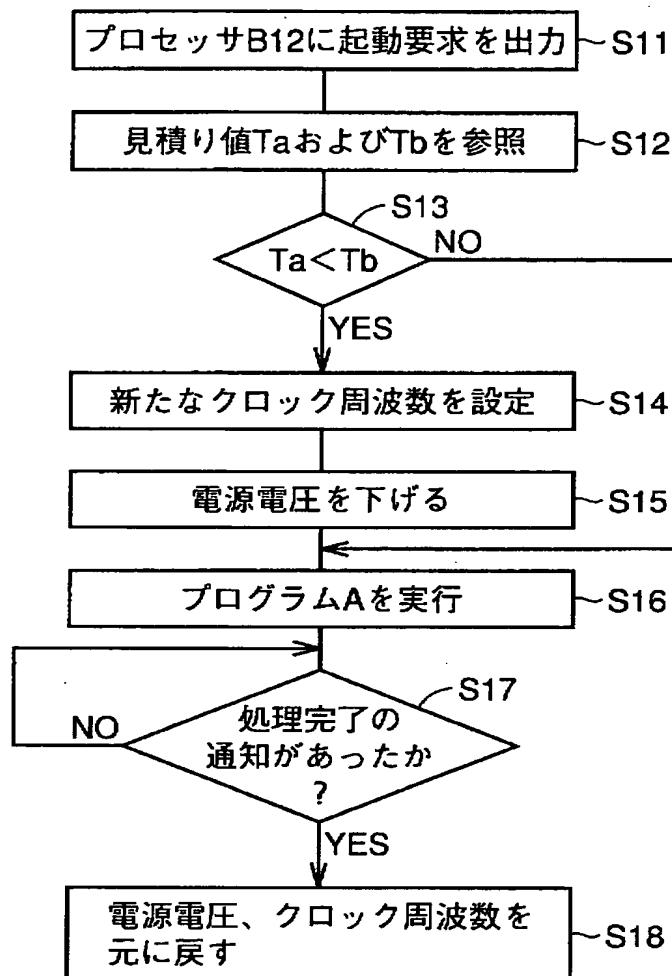
【図1】



【図2】



【図3】



【図4】

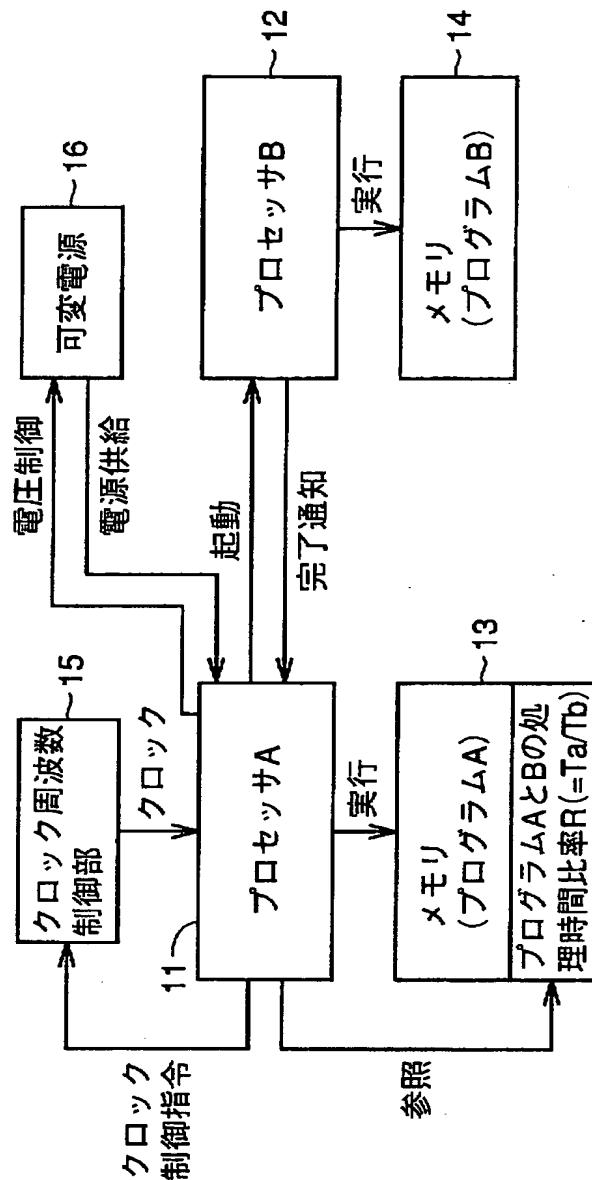
```

main_procedure()          /* プログラムAのメイン処理部 */
{
    B_is_done=0;          /* プログラムBを起動する。処理時間見積りはTb */
    invokeB();             /* クロック周波数をTa/Tbを限度として遅くする。*/
    set_clock_frequency(Ta/Tb); /* テーブルを参照して電圧を下げる */
    set_power(table);      /* プログラムAで行うべき処理。処理時間見積りはTa */
    do_something();        /* プログラムBの完了を待つ */
    while (B_is_done)      /* プログラムAの割り込み処理ループ */
        /* do nothing */;
}

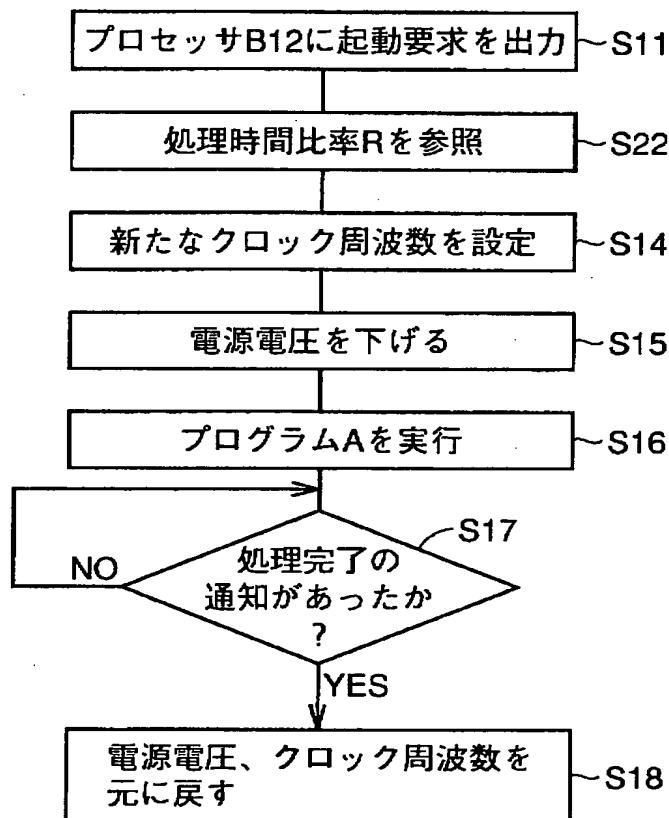
interrupt_from_B()        /* プログラムAの割り込み処理ループ */
{
    restore_power();        /* 電圧を元に戻す */
    restore_clock_frequency(); /* クロック周波数を元に戻す */
    B_is_done=1;            /*
}

```

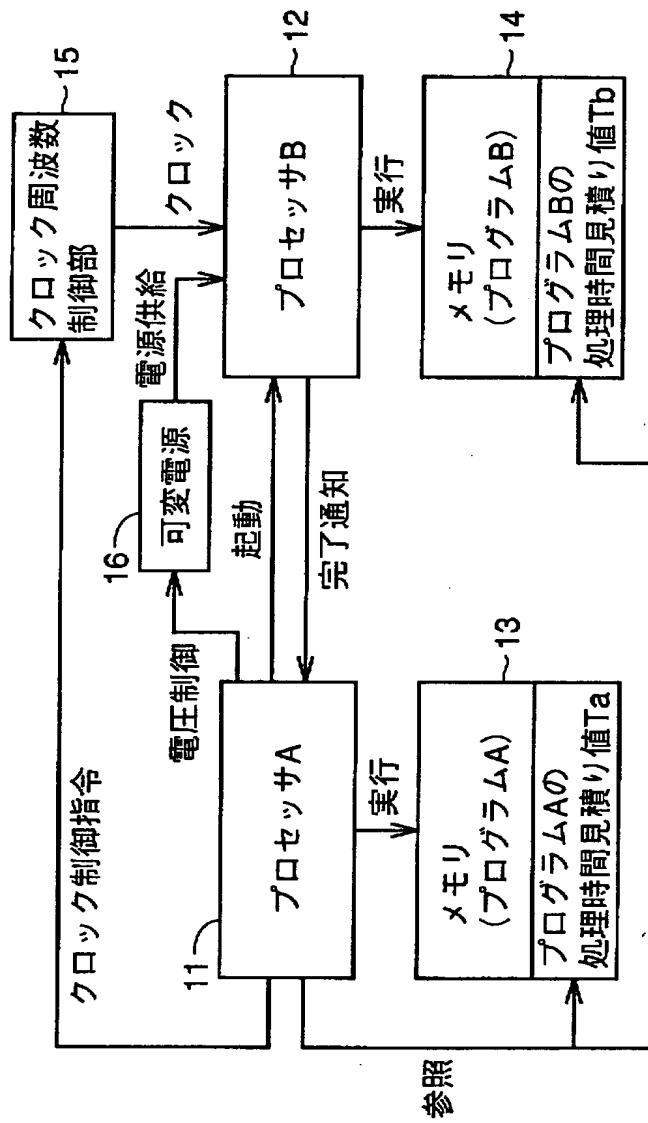
【図5】



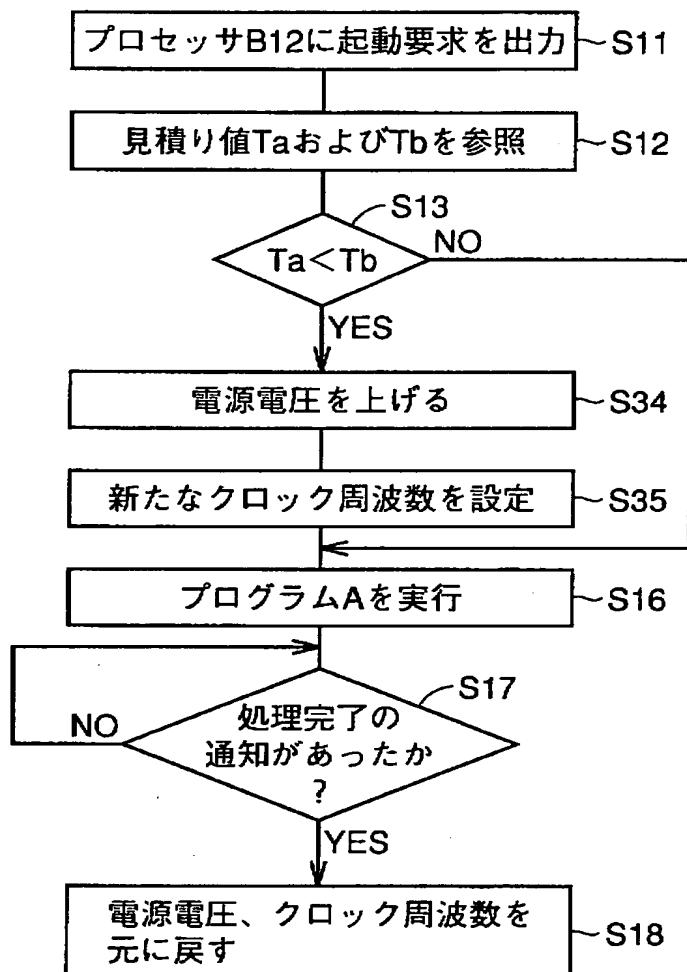
【図6】



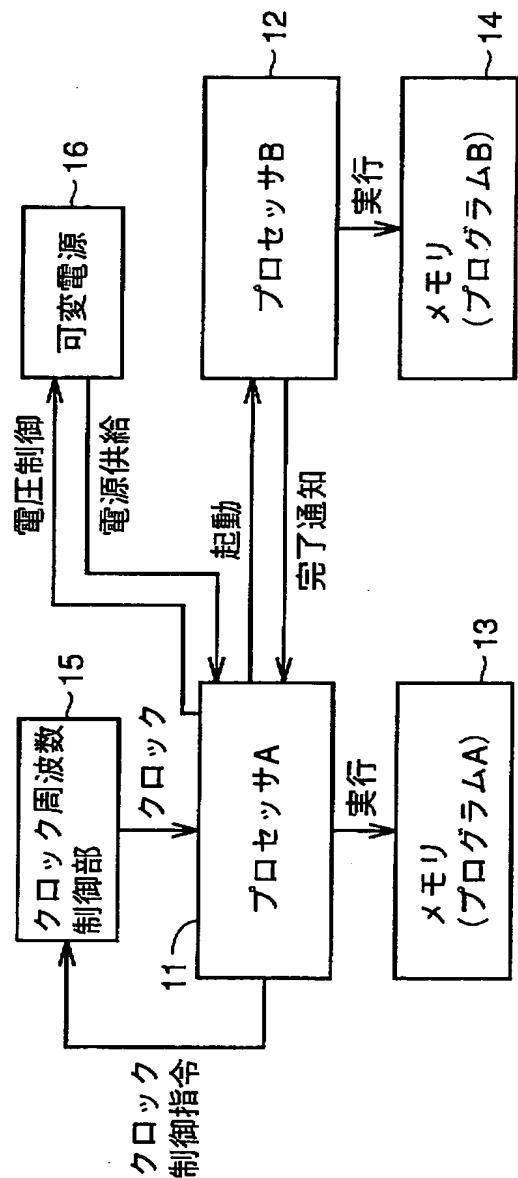
【図7】



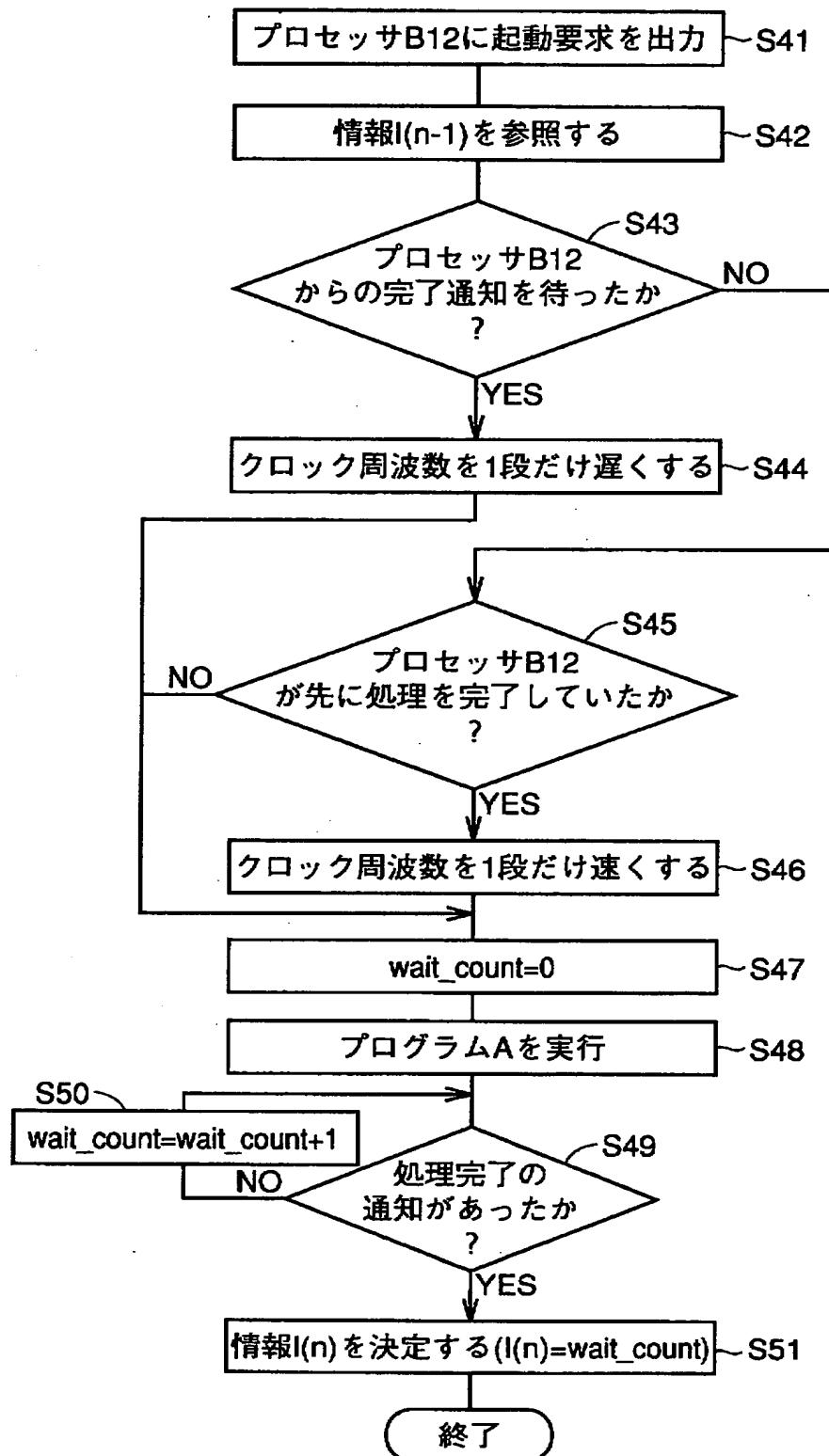
【図8】



【図9】



【図10】



【図11】

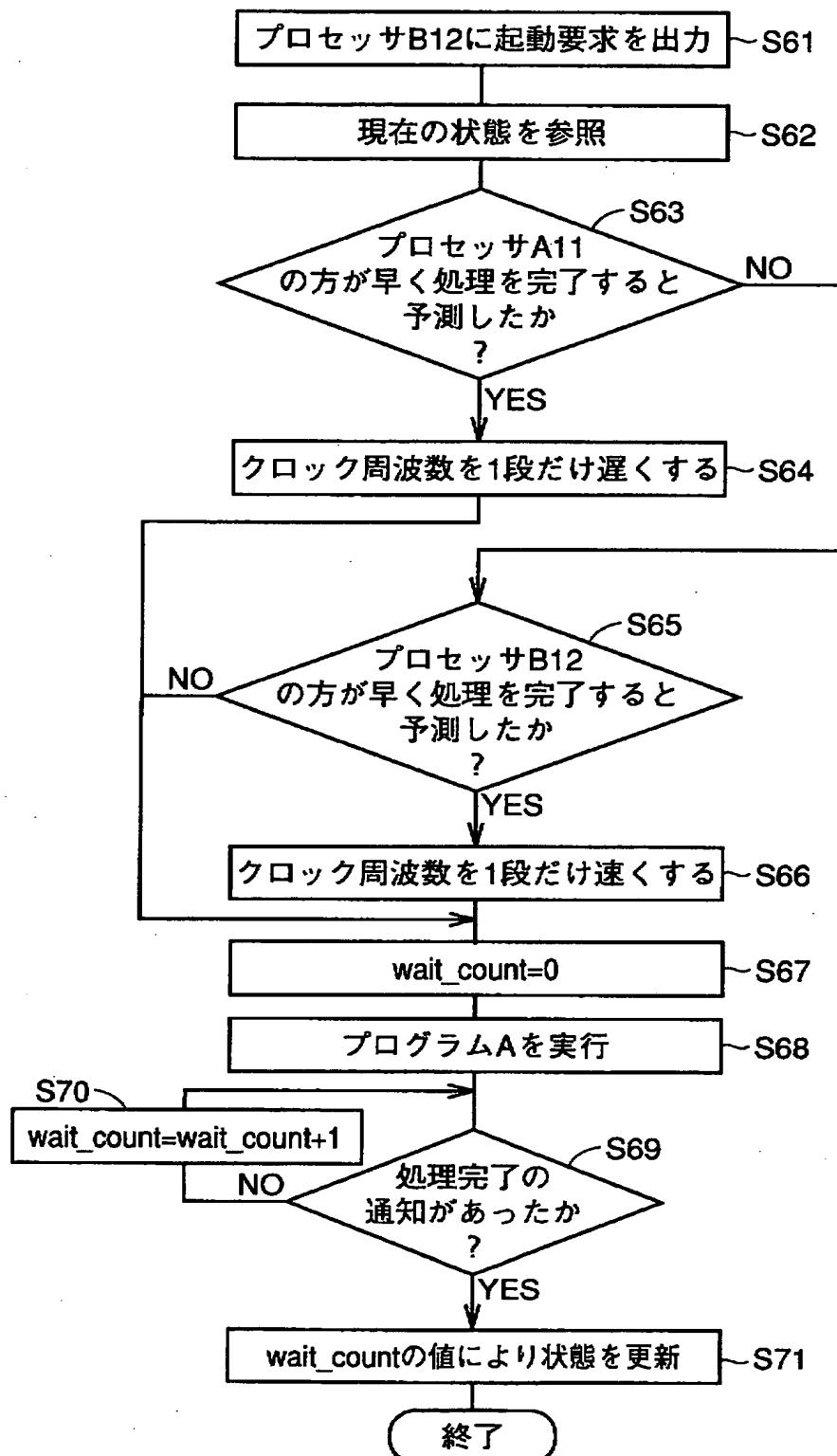
```

main_procedure()      /* プログラムAのメイン処理部 */
{
    B_is_done=0;
    /* プログラムBを起動する。 */
    if(wait_count>WAIT_LIMIT) /* 前回の処理にてプログラムBの完了を待った場合には */
        /* クロック周波数を1段だけ遅くする。 */
        /* プログラムBの方が早く完了した場合には */
        /* クロック周波数を1段だけ速くする。 */
        /* プログラムAとプログラムBが同じくらいに完了した場合には */
        /* 何もせずに現在のクロック周波数を維持する */
    wait_count=0;
    do_something();          /* プログラムAで行うべき処理。 */
    while(B_is_done==0)      /* プログラムBの完了を待つ */
        wait_count++;
}

interrupt_from_B()
{
    B_is_done=1;
}

```

【図12】



【図13】

```

main_procedure()          /* プログラムAのメイン処理部 */
{
    B_is_done=0;
    invokeB();           /* プログラムBを起動する。 */

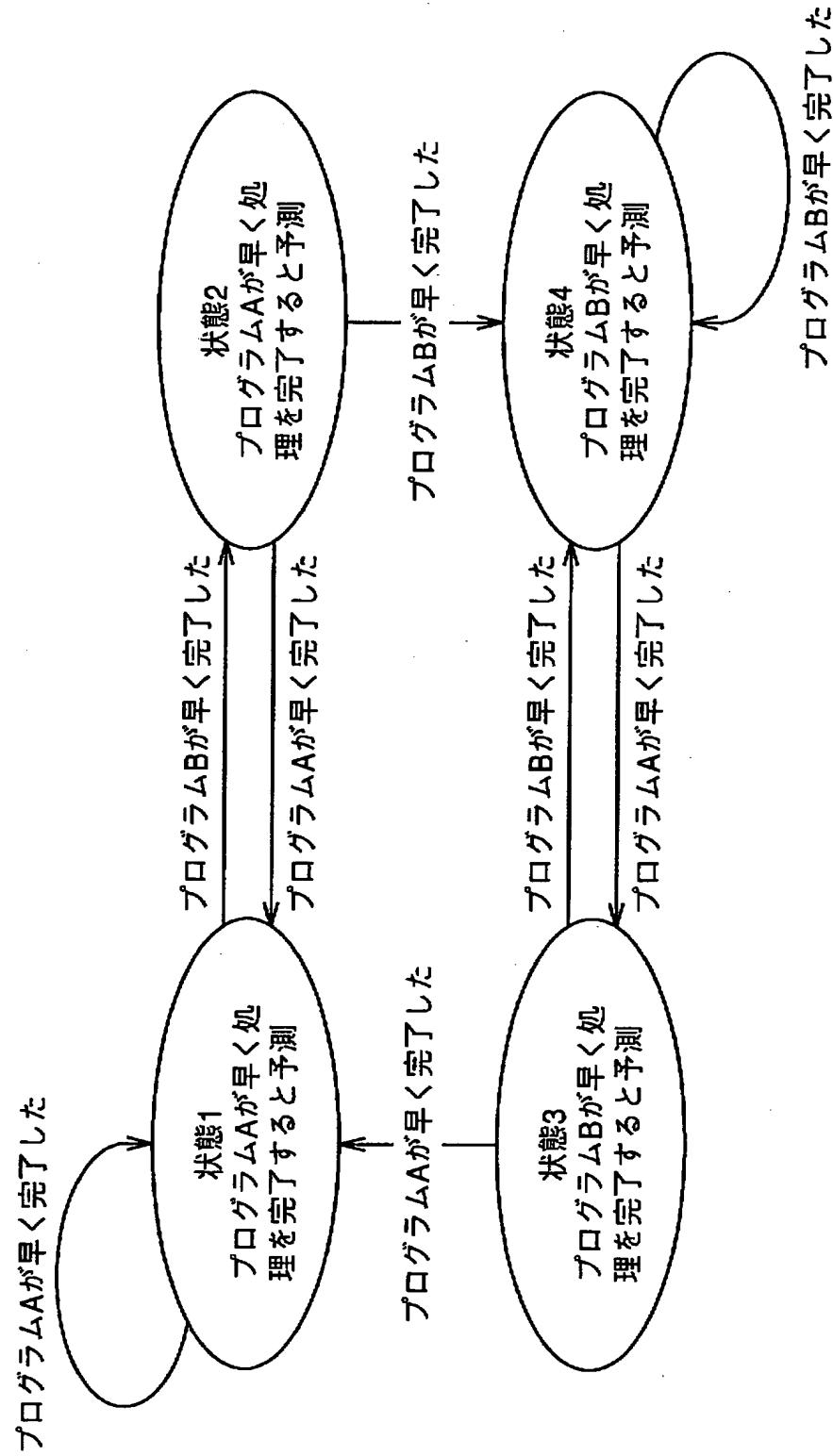
    予測値=現在の状態から予測値を得る;
    if(予測値==プログラムAが早く処理を完了すると予測)
        set_clock_faster(); /* クロック周波数を1段だけ遅くする。 */
    else(予測値==プログラムBが早く処理を完了すると予測)
        set_clock_slower(); /* クロック周波数を1段だけ速くする。 */
    else/* プログラムAとプログラムBが同じくらいに完了すると予測*/
        /* do nothing */;
    wait_count=0;

    do_something();        /* プログラムAで行うべき処理。 */
    while(B_is_done==0)
        wait_count++;
        wait_countの値より、状態を更新する;
    }

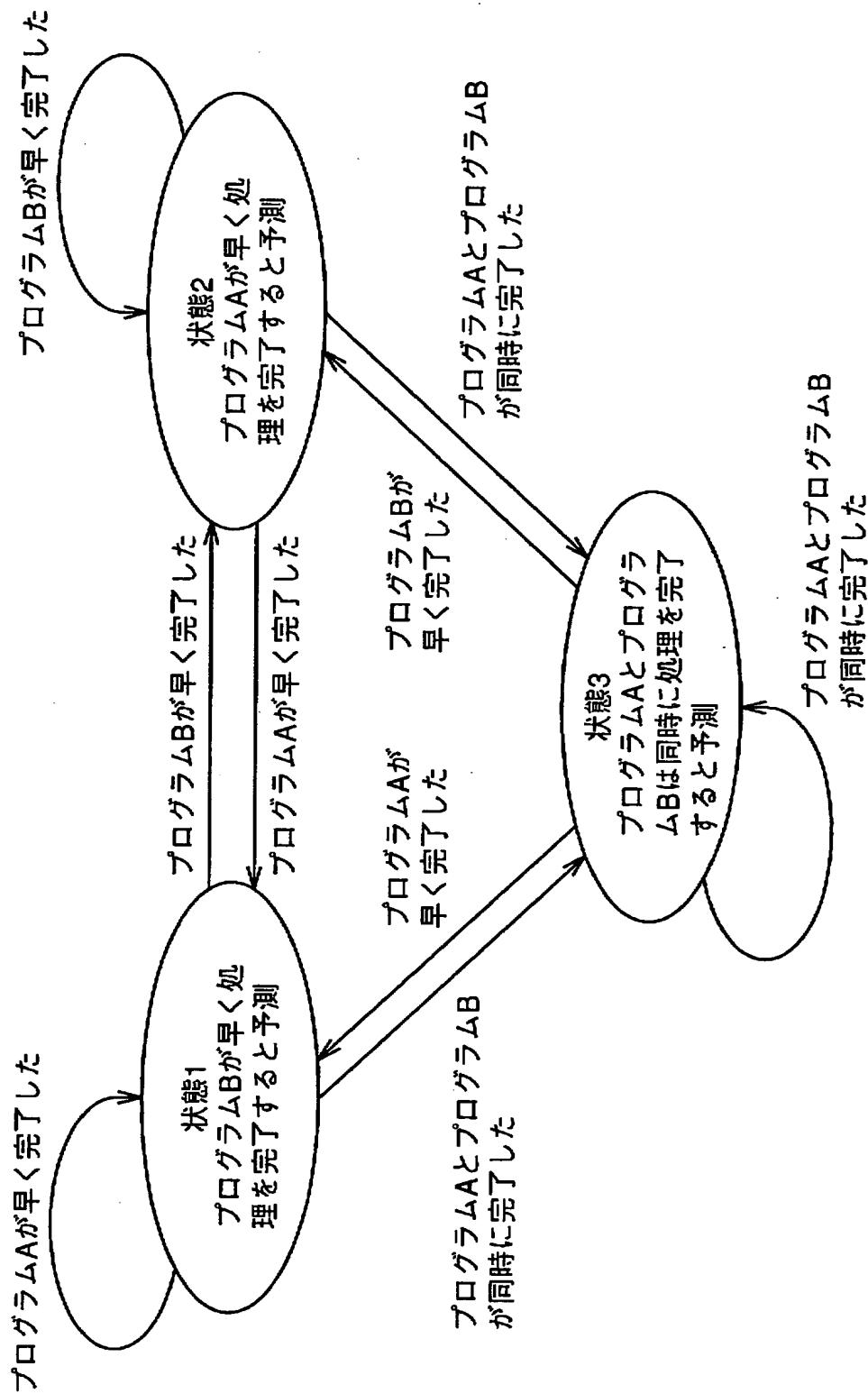
    interrupt_from_B()    /* プログラムAの割り込み処理ルーチン */
    {
        B_is_done=1;
    }
}

```

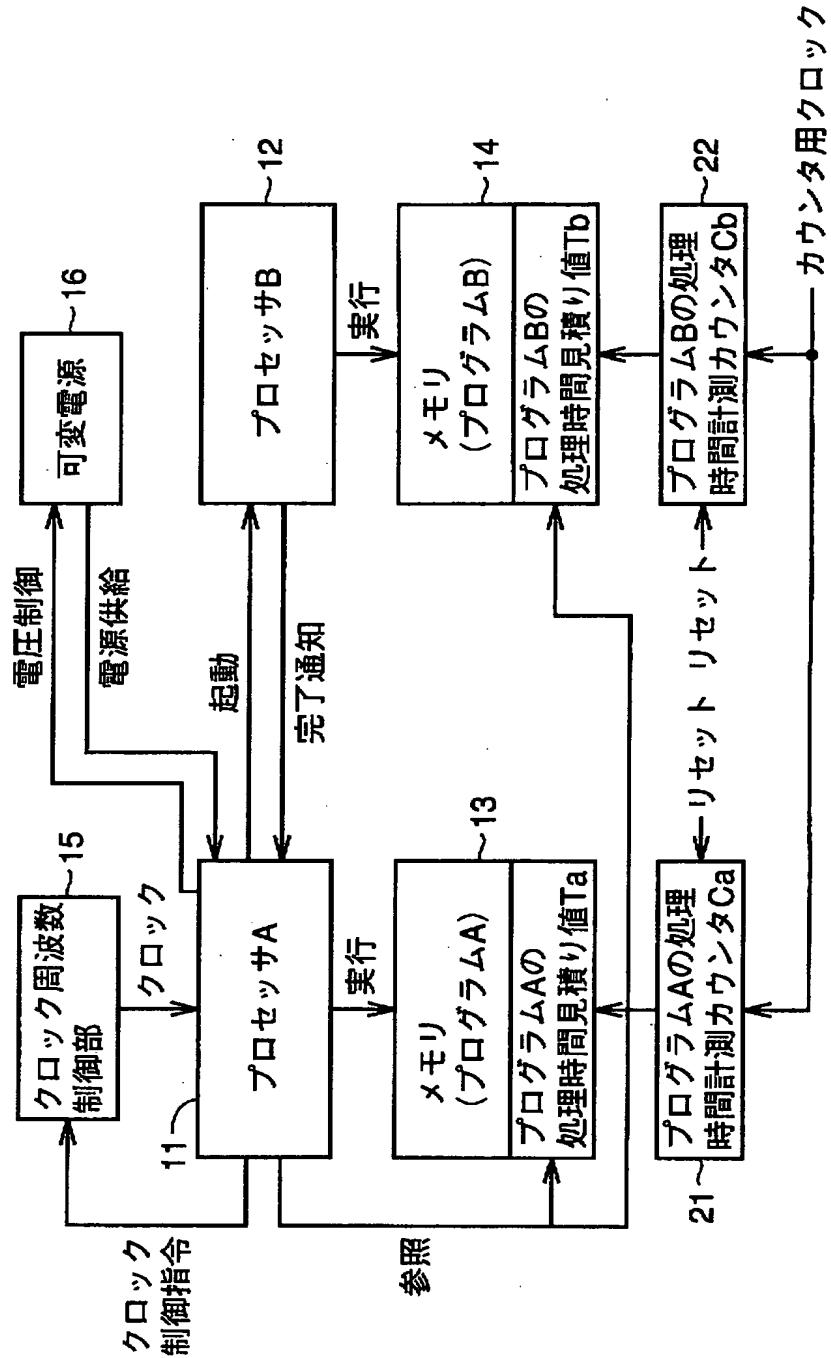
【図14】



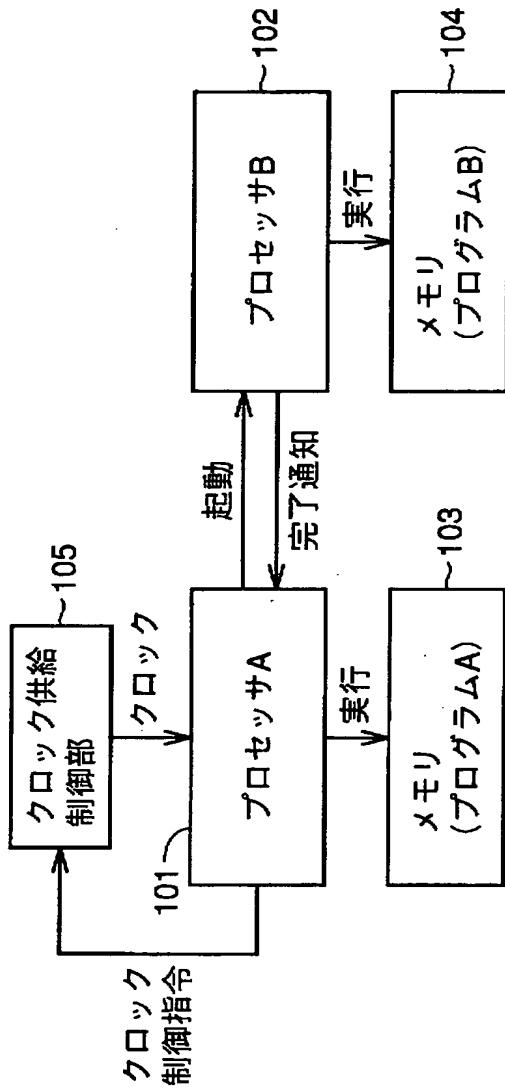
【図15】



【図16】



【図17】



【書類名】 要約書

【要約】

【課題】 消費電力を大きく削減することが可能なマルチプロセッサシステムを提供すること。

【解決手段】 プロセッサA11は、プログラムAの処理時間見積値TaとプログラムBの処理時間見積値Tbとの比率に応じて、クロック周波数制御部15を制御してプロセッサA11に入力されるクロックの周波数を変更する。また、プロセッサA11は、プロセッサA11に入力されるクロックの周波数に応じて可変電源16を制御して、プロセッサA11に供給される電源電圧を変更する。したがって、プロセッサA11の消費電力を大きく削減することが可能となる。

【選択図】 図1

出願人履歴情報

識別番号 [000006013]

1. 変更年月日 1990年 8月24日

[変更理由] 新規登録

住 所 東京都千代田区丸の内2丁目2番3号

氏 名 三菱電機株式会社